

## Fizz-Buzz-Fizzbuzz

```
def fizz_buzz(n):
    fizz = 0
    buzz = 0
    fizz_buzz = 0
    for i in range(1, n+1):
        if i % 3 == 0:
            fizz += 1
            if i % 5 == 0:
                fizz_buzz += 1
        if i % 5 == 0:
            buzz += 1
    return fizz, buzz, fizz_buzz

if __name__ == '__main__':
    import sys
    for line in sys.stdin:
        print(' '.join([str(x) for x in fizz_buzz(int(line))]))
```

# Od noviga lejta dni

```
def days(year):
    n = (2024 - year + 1) * 365
    n += len([x for x in range(year, 2025) if x % 4 == 0])
    return n

if __name__ == '__main__':
    import sys
    for year in sys.stdin:
        year = int(year)
        print(days(year))
```

## Prva pred drugimi

```
def first_before_second(ch, s):
    f_max = s.rfind(ch[0])
    c_min = s.find(ch[1])
    return f_max < c_min

if __name__ == '__main__':
    import sys
    for l in sys.stdin:
        ch = l[:2]
        s = l[2:]
        print(first_before_second(ch, s))
```

## Kakšno je zaporedje

```
def diff(nums):
    return list(map(lambda x, y: x - y, nums[1:], nums[:-1]))

def seq_lin_qua_cub(nums):
    lst = ["linear", "quadratic", "cubic"]
    for i in range(3):
        nums = diff(nums)
        if len(set(nums)) == 1:
            return lst[i]
    return False

if __name__ == '__main__':
    import sys
    for nums in sys.stdin:
        nums = [int(x) for x in nums.split()]
        print(seq_lin_qua_cub(nums))
```

# Kaprekarjev par

```
def kaprekar(x):
    n = len(x)
    mini = ''.join(sorted(x))          # uredi številke po velikosti v premem
    maxi = ''.join(sorted(x, reverse=True)) # in obratnem vrstnem redu
    x = str(int(maxi) - int(mini))      # x je razlika
    x = x.zfill(n)                     # doda vodilne ničle, da dolžina ostane n
    return x

def kaprekar_(x):
    visited = {x}
    while True:
        x = kaprekar(x)
        if x in visited:
            break
        visited.add(x)
    return x

def kaprekar_cycle(n):
    nn = [kaprekar_(n)]
    while True:
        m = kaprekar(nn[-1])
        if m == nn[0]:
            break
        nn.append(m)
    return ' '.join(nn)

if __name__ == '__main__':
    import sys
    for n in sys.stdin:
        n = int(n)
        print(kaprekar_(str(n)))
```